REPORT DOCUMENTATION PAGE	*Form Approved* *OMB No. 0704-0188*

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From - To)*
17-MAR-2004	Conference Proceedings, (refereed)	

4. TITLE AND SUBTITLE
POS Polyline Smoothing: Reduction Of Polyline Vertices

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER
62435N

6. AUTHOR(S)
GEARY LAYNE   MARLIN GENDRON   MAURA C LOHRENZ

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER
74-6636-C4

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Naval Research Laboratory
Marine Geoscience Division
Stennis Space Center, MS 39529-5004

8. REPORTING ORGANIZATION REPORT NUMBER
NRL/PP/7440--04-1011

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Office of Naval Research
800 N. Quincy Street
Arlington, VA 22217

10. SPONSOR/MONITOR'S ACRONYM(S)
ONR

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release,distribution is unlimited

13. SUPPLEMENTARY NOTES

14. ABSTRACT
The POS polyline smoothing algorithm was developed to reduce the needed storage and rendering complexity of polylines by the removal of vertices with two goals in mind. First was to define a single algorithm that would produce a "good enough" result with varying characteristics, which are user defined. The concept of "good enough" is built on the trade off of time vs. precision, where the best result takes the longest time and the quickest maybe less than desirable form. The second goal was to incorporate surrounding data into the set of control factors. To accomplish this, the concept of Points of Significance (POS) was developed. POS can be a group of individual points, or the point that represents more complex shapes or regions. The complete set of POS is divided into subsets by the polyline, and by maintaining these subsets the algorithm insures that the polyline maintains a proper relationship with the surrounding data. It is the use of POS that makes this algorithm so powerful. The smoothing is complete after a maximum number of successive passes are made through the polyline, or no additional removals can be made without violating the control factors.

15. SUBJECT TERMS
POS polyline smoothing algorithm

16. SECURITY CLASSIFICATION OF: a. REPORT	b. ABSTRACT	c. THIS PAGE	17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON / 19b. TELEPHONE NUMBER *(Include area code)*
Unclassified	Unclassified	Unclassified	Unlimited	7	Geary Layne / 228-688-4693

POS Polyline Smoothing: Reduction of Polyline Vertices

Geary Layne, Naval Research Laboratory
Marlin Gendron, Naval Research Laboratory
Maura Lohrenz, Naval Research Laboratory

Abstract

The POS polyline smoothing algorithm was developed to reduce the needed storage and rendering complexity of polylines by the removal of vertices with two goals in mind. First was to define a single algorithm that would produce a "good enough" result with varying characteristics, which are user defined. The concept of "good enough" is built on the trade off of time vs. precision, where the best result takes the longest time and the quickest maybe less than desirable form. The second goal was to incorporate surrounding data into the set of control factors. To accomplish this, the concept of Points of Significance (POS) was developed. POS can be a group of individual points, or the point that represents more complex shapes or regions. The complete set of POS is divided into subsets by the polyline, and by maintaining these subsets the algorithm insures that the polyline maintains a proper relationship with the surrounding data. It is the use of POS that makes this algorithm so powerful. The smoothing is complete after a maximum number of successive passes are made through the polyline, or no additional removals can be made without violating the control factors.

Introduction

In 2000, the authors designed and implemented a computer aided detection (CAD) algorithm capable of detecting bottom objects, or clutter, in sidescan imagery (SSI). A unique latitude and longitude (LAT/LON) position was recorded for each detected object. The authors then wrote a clustering algorithm, which is currently patent pending, that clusters all points of significance (POS) into bound geographic polygons. The area inside each polygon is computed and a clutter density is determined (figure 1).

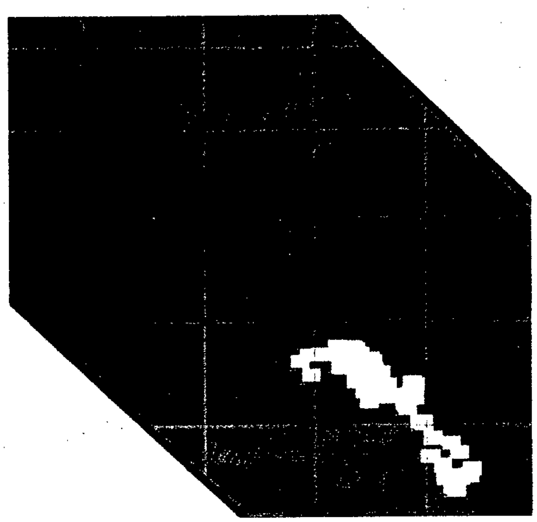

Figure 1 – Polygons produced by the NRL clustering algorithm

The ordered vertices of the polygons are determined and passed to a geographical information system (GIS) application that renders them. The clustering algorithm is a single pass method and computationally fast, but it produces polygons with many sides. Thus, the GIS program must draw many small line segments to render the polygons, which impacts the overall processing time. Furthermore, when drawn, the polygons are not esthetically pleasing to the eye. To correct this problem, NRL designed, developed and implemented the polygon-smoothing algorithm described in this paper. Figure 2 shows how the algorithm might smoothed the polygons from figure 1.

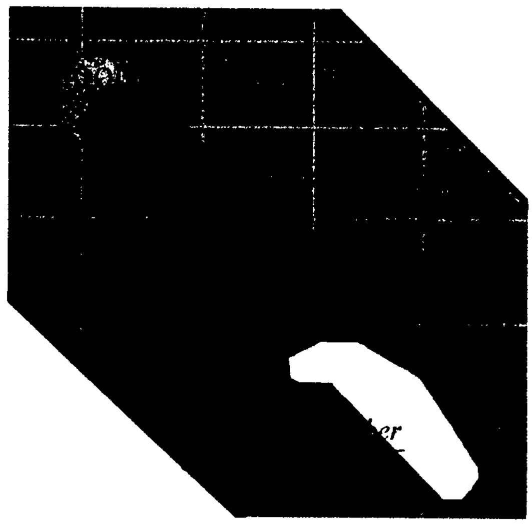

Figure 2 – Polygons after POS polyline smoothing algorithm is applied

The clustering algorithm is greatly improved by the polygon-smoothing algorithm. This paper describes the polygon-smoothing algorithm in detail and gives examples. Currently, the authors are modifying the smoothing algorithm to make it more generic, robust, and powerful. The new algorithm, to be described in a future paper, will also be capable of smoothing non-bounding groups of line segments.

Background

The CAD algorithm developed by NRL determines the LAT/LON positions of POSs, which in this case represent features on the bottom of the seafloor detected in SSI. To determine the density of the objects on the seafloor, NRL developed a clustering algorithm. In general, clustering is the process of grouping like objects together. Clustering methods can be classified as either hierarchical or non-hierarchical, as summarized in Barnard (1996) and Downs (2001). For this specific application, the characteristic that defines the similarity of the objects is their geographical location

(figure 3). NRLs non-hierarchical clustering algorithm, in addition to clustering, bounds the objects into geographic polygons.

The algorithm bounds the individual POSs and unions them into clusters. A special algorithm is then applied that traces the exterior of the cluster and determines the vertices, which define a polygon. This method, although extremely fast, produces polygons with a substantial number of sides. The overall goal of the project was to compute a density for these polygons and maintain the efficiency of the clustering process. NRL needed an algorithm that would smooth the polygons by reducing the number of sides while not decreasing their density. The algorithm also must not significantly impact the overall processing time.

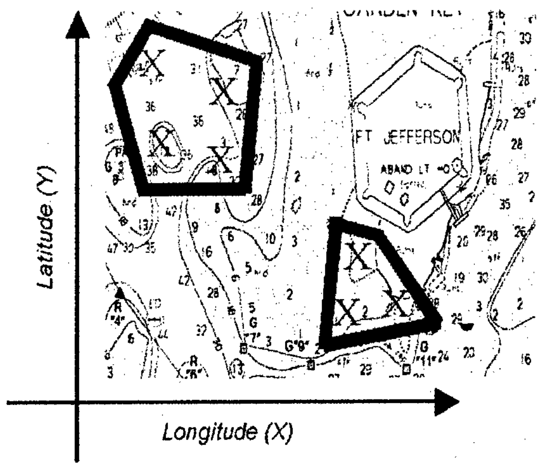

Figure 3 - Clustering

Polygon-Smoothing Algorithm

The clustering algorithm determines the POSs that fall within the clusters, and the tracing algorithm determines a clockwise ordered list of vertices. The list and the points are then passed to the smoothing algorithm. The smoothing algorithm traverses the list iteratively until one of the stopping conditions (described below) is met. Upon completion, the algorithm returns a modified list of vertices and the area which will be equal to or less than the starting area.

During each iteration, or one complete traversal of the list, a decision to drop vertices is made based on the current vertex. The current vertex and the following three vertices are considered independently of the rest. There are three steps in the decision-making process: 1) determine if the next vertex can be removed, 2) determine if the next two vertices can be removed, and 3) determine which of the available removal options is most desirable.

Considered the following example, noting that the interior of the polygon is below the polyline (figure 4).

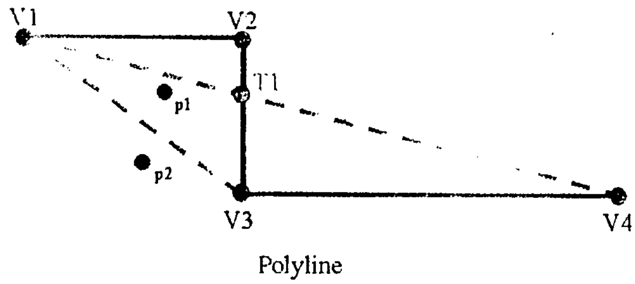

Figure 4 – Polyline Smoothing Example

The first decision (the removal of vertex V2) is made by determining whether any POSs exist in the interior of the triangle defined by V1, V2, and V3. Thus if the point, p1, is an element of the POSs, the removal of V2 would not be allowed, because the resulting polygon would no longer contain all the POSs. If p1 did not exist, the removal would be allowed. The second decision (to allow the removal of V2 and V3 together) is based on the line connecting V1 and V4. As a result of the line intersecting V2 and V3 at T1, the decision is broken into two components.

Since the triangle defined by V1, V2 and T1 could potentially contain POSs, a check is required; however, no check is needed for the triangle T1, V4, and V3 because its interior is always exterior to the polygon.

The third decision is based on the area of the resulting polygon. The removal that results in the greatest reduction in area is selected. In figure 4, the removal of V2 alone would not be allowed because of p1; however, the removal of V2 and V3 together would be allowed if the reduction in area by triangle V1, V2, and T1 and increase in area by triangle T1, V4, and V3 results in a net decrease in the area of the polygon.

Now considered the example polyline in figure 5:

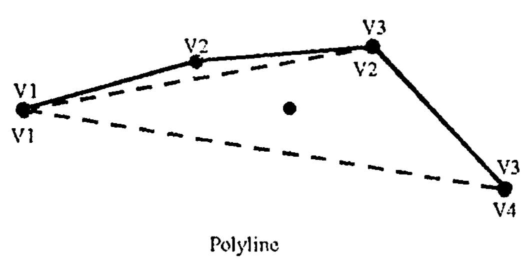

Figure 5 – Non-Intersecting Line Segment

Similarly, step 1 is to determine if V2 can be removed. Since there are no point in the interior of the triangle V1, V2, and V3, this would be allowed. Since the line segment connecting V1 and V4 does not intersect the line segment connecting V2 and V3, step two (the decision to remove V2 and V3) can be reduced to the consideration of whether or not the triangle V1, V3, and V4 contains any POSs.

In this example, step three is straight forward since the triangle V1, V2, and V3 does not contain a POS and the triangle V1, V3, and V4 does. Because of this, V2 is removed. If, however, there were not a point inside V1, V3, and V4, then both V2 and V3 would be removed, since this option would reduce the area the most.

Finally, two special cases occur that are worthy of mention. If V1, V2 and V3 are co-linear (top of figure 6), V2 can be immediately removed. If the removal of V2 (bottom of figure 6) is determined to be acceptable, and V1, V3, and V4 are co-linear, the removal of both V2 and V3 is automatic.

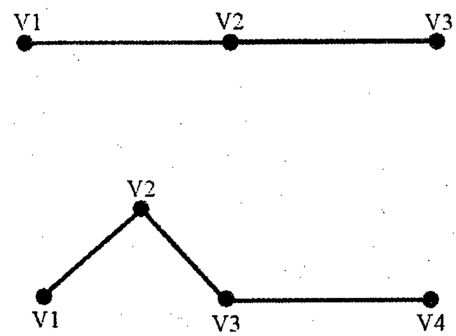

Figure 6 – Special Cases

After performing the above three steps on any configuration and removing the appropriate vertices, the next vertex in the list becomes the current vertex and the process continues. The smoothing ends when either 1) a predetermined number of iterations is made, 2) a complete iteration is made with no removals, or 3) the polygon is reduced to three vertices.

Geographic Bitmaps

The smoothing algorithm relies heavily on geo-spatial bitmap (GB) for improved computational speed. Bitmaps are two-dimensional binary structures in which bits are turned on (set) or off (cleared), and the row and column of each bit gives it a unique position. This concept is extended to construct GB's, where every bit represents a unique location in a coordinate system at a given resolution (figure 7). A set bit denotes that data exists at a specific coordinate. Although the GB is defined for the entire coordinate system at a given resolution, memory is only allocated dynamically when groups of spatially close bits are set. This makes the GB a fast and compact data structure (Gendron, et al., 1997).

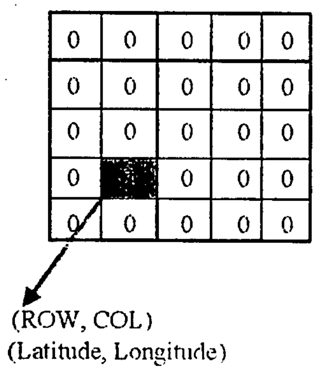

Figure 7 – Geographic Bitmap

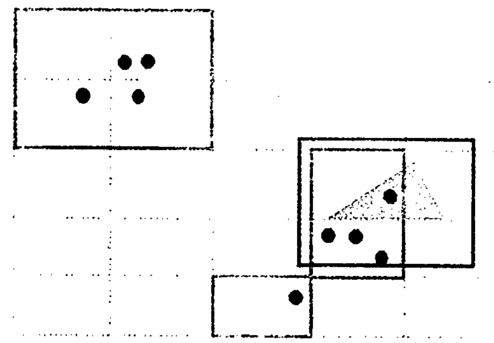

Figure 8 - Anding in GB Space

Logical operations can be performed between GB's of differing sizes and starting locations (figure 8). All POSs are represented by a single GB (light gray boundary of figure 8, where memory is allocated only for the regions outlined in red). Each triangle defining the result of removing a vertex is represented by a small GB (blue area of figure 8), where all bits representing the triangle are set. The logical ANDing between the POS GB and the triangle GB is driven by the typically smaller triangle GB, which results in a very efficient operation. If the resulting GB contains any set bits, it is immediately known that a POS lies in the interior of the triangle and thus the vertex under consideration may not be removed.

Conclusion

The use of GB's and corresponding logical operations, renders the smoothing algorithm vary fast and memory-efficient, since geographical comparisons and Euclidian distance calculations are not required. The algorithm takes the complex problem of smoothing and breaks it down into manageable components.

Although this algorithm was designed to smooth polygons, the process considers only segments of the polygon at a time. It is currently being adapted to function on arbitrary polylines and POSs. The meaning of the term POS will be expanded, and because the polylines no longer bound the POSs a new process for determining regions, rather than the interior of the polygon, will be introduced in a future paper.

Acknowledgments

This work was funded by the Office of Naval Research through the Naval Research Laboratory under Program Element 62435N. The mention of commercial products or the use of company names does not in any way imply endorsement by the U.S. Navy. Approved for public release; distribution is unlimited.

References

J. M. Barnard (1996). *Agglomerative Hierarchical Clustering Package from Barnard Chemical Information, Ltd.* Presented at the Daylight EUROMUG Meeting, Basel, Switzerland,17Dec.URL:http://www.daylight.com/meetings/emug96/john_barnard/emug96.html.

G. Downs (2001). *Clustering in Chemistry.* Presented at the MathFIT workshop, Belfast, 27 April. URL: http://www.qub.ac.uk/ivs/mvl/geoff-downs.pdf.

M. L. Gendron, P. B. Wasco, M.E. Trenchard, M.C. Lohrenz, L.M. Riedlinger, M.J. Mehaffey (1997). "Moving Map Composer," Naval Research Laboratory, US Patent No. 6,218,965 B1, 1-29.